

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



**PROYECTO FIN DE CARRERA**

## **OPTIMIZACIÓN AUTOMÁTICA DEL PERFIL DE ÁLABE EN TURBINAS SUBMARINAS**

Realizado por: **Víctor Rodríguez Rodríguez**

Titulación: Ingeniería Industrial, especialidad en Tecnologías Energéticas

Mediante programa de movilidad ERASMUS

Universidad de destino: Technische Universität Kaiserslautern (Alemania)

Tutor: Dr. Harald Roelawski

Coordinador académico: Miguel Ángel Moscoso Castro

Cotutor en UCIIM: Miguel Ángel Moscoso Castro

Calificación obtenida: 10, sobresaliente

Fecha de finalización: septiembre 2012

## OBJETIVOS

El objetivo principal de este estudio es encontrar el diseño óptimo del perfil de una turbina oceánica que sea capaz de generar la máxima fuerza de sustentación posible. Dicho máximo valor supondrá asimismo un momento máximo sobre el rotor, y en última instancia, una optimización de la potencia generada.

Para lograr este objetivo principal, se requiere primeramente satisfacer una serie de objetivos parciales:

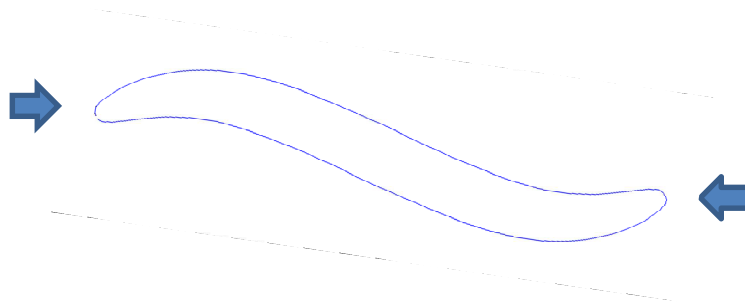
- A partir de un código en lenguaje C++ previo que se encarga de generar el perfil del álabe, adaptar dicho código de tal manera que acepte variables que puedan ser introducidas por teclado, pudiendo manualmente variar la geometría a voluntad. A su vez, permitirá devolver las variables de salida que se requieran para los siguientes procesos.
- Utilizando el software de CFD *ANSYS Fluent*, y las herramientas de diseño y mallado, importar la geometría generada en C++, y simular numéricamente el problema dadas unas determinadas condiciones de contorno. Extraer el valor de la fuerza de sustentación.
- Al mando de todos estos procesos se encuentra *MATLAB*, mediante diversas funciones y *scripts*, a través de los cuales las variables de entrada pueden ser definidas, la geometría para cada caso actualizada e importada en el software de CFD, y por último, tras la simulación, cada nuevo valor de fuerza de sustentación almacenado.
- Ser capaz de realizar la optimización utilizando la herramienta *Optimization Toolbox* disponible en *MATLAB*. Dicha herramienta, mediante los determinados algoritmos que posee, es capaz de minimizar/maximizar el valor de una determinada función objetivo, dadas unas determinadas restricciones.

## DESCRIPCIÓN DEL PROBLEMA

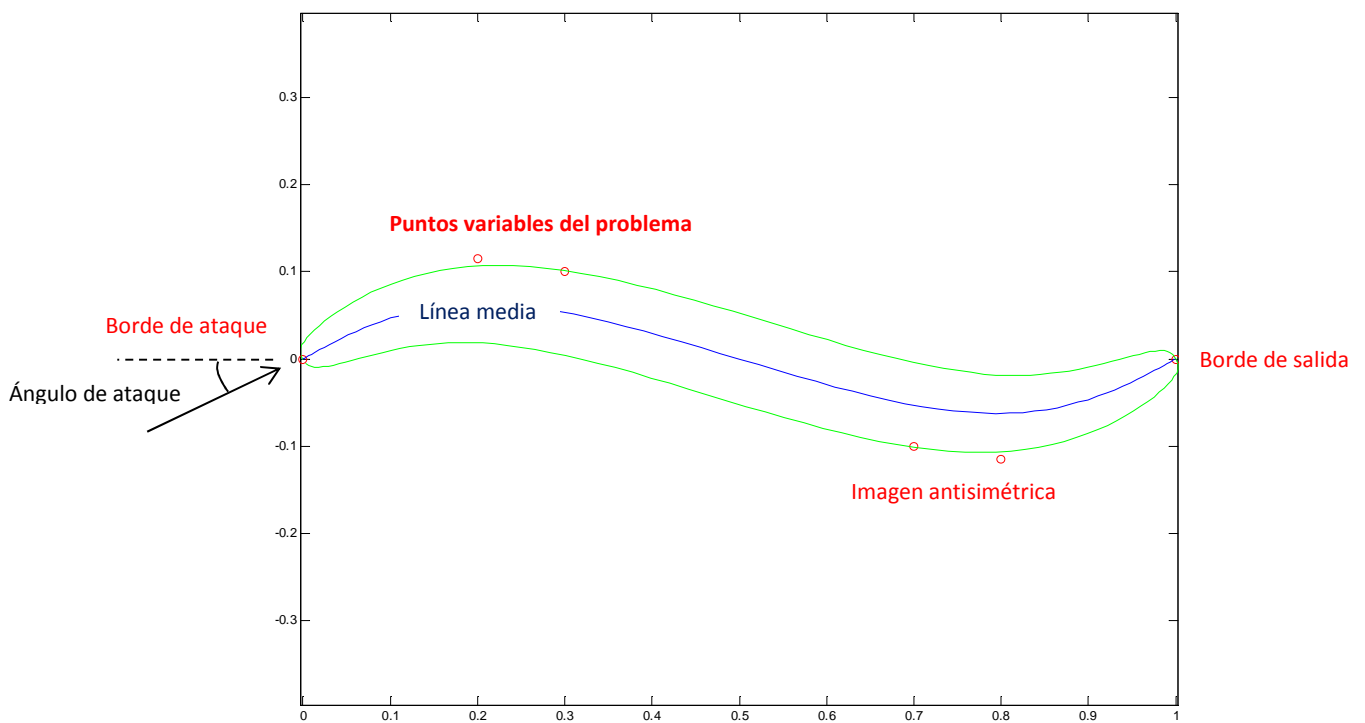
La característica distintiva del tipo de perfil utilizado en este estudio es su simetría, con lo que es capaz de funcionar en el caso de corrientes cíclicas que cambien de sentido (por ejemplo, en las corrientes derivadas de las mareas). A diferencia de los perfiles convencionales, donde la corriente sólo puede incidir en un sentido, puesto que las pérdidas se verían drásticamente incrementadas si la corriente incidiese desde el borde de salida.



Sin embargo, en el caso del nuevo diseño, simétrico, ambos bordes pueden actuar como borde de ataque o de salida según el sentido de la corriente incidente.



El perfil está formado por una línea media y una distribución de grosor. La línea media está basada en una *spline* de Camber. Los dos puntos, cuyas coordenadas son precisamente las variables de entrada del problema, controlan la curvatura de la *spline*. El objetivo de este estudio es encontrar la combinación de coordenadas de dichos puntos que den como resultado un perfil que presente el mayor valor de fuerza sustentadora.



## **PRODEDIMIENTO**

### ***GENERACIÓN DE LA GEOMETRÍA DEL PERFIL EN C++***

En primer lugar se modificó el código en C++ de tal forma que el nuevo código permitiese definir las variables de entrada desde la ventana de comando, pudiendo variar los puntos que definen la geometría manualmente.

### ***GEOMETRÍA EN MATLAB***

El ejecutable creado tras el depurado del nuevo código de C++, al que se accede en MATLAB mediante el comando DOS, se encarga de generar los puntos que definen el perfil en archivos *.dat*, según sean las coordenadas de las variables introducidas.

### ***SIMULACIÓN DEL FLUJO EN ANSYS WORKBENCH***

El flujo de trabajo es controlado en todo momento por un script en lenguaje Python que se encarga de importar la nueva geometría y actualizar el resto del problema en *Fluent*.

#### **Geometría**

Por medio del Design Modeler de ANSYS, la geometría para cada perfil diferente puede ser importada para la resolución del campo fluido definido alrededor del mismo.

#### **Mallado**

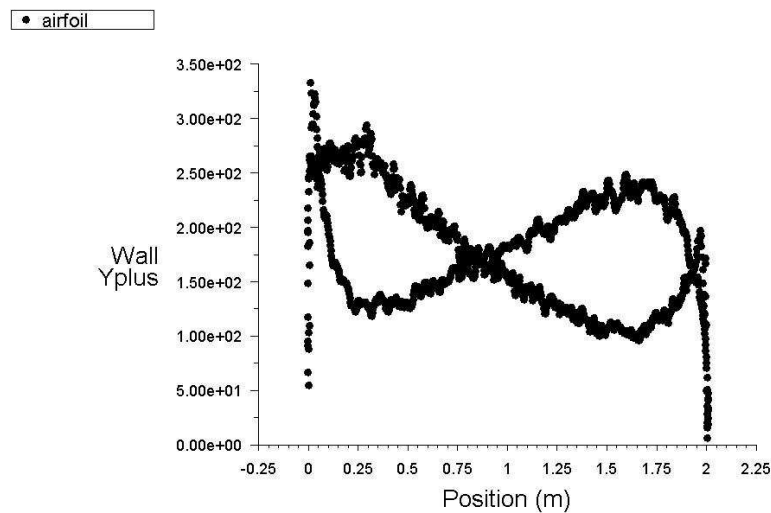
El criterio de mallado se toma de tal forma que el número de divisiones, esto es, el refinamiento de la malla, sea mayor conforme se aproxima a la zona donde se encuentra el perfil. De esta forma, la malla será capaz de resolver de manera más fiable el campo fluido alrededor del mismo.

#### **Problema en *Fluent***

Puesto que se trata de un flujo no compresible (agua), se elige un modelo basado en presión (*pressure-based solver*). Las condiciones de contorno son, velocidad de entrada del flujo 2 m/s con ángulo de ataque 5°, la presión manométrica a la salida del campo fluido es cero, y el perfil representa una pared estacionaria.

Teniendo un valor de número de Reynolds de orden  $10^6$ , y considerando que para poder resolver la capa límite sería necesaria una malla extremadamente refinada, con un coste computacional excesivo, y siendo necesario un elevado número de iteraciones para resolver cada caso, se opta por el modelo de turbulencia *k-epsilon*. Se controla el valor de  $y^+$  para comprobar que en efecto la malla utilizada es suficientemente fiable para resolver el problema cerca de la pared ( $30 \leq y^+ \leq 300$ ) para función de pared estándar.

Teniendo en cuenta el número de iteraciones hasta convergencia, y la variabilidad de los resultados según el orden de magnitud de los residuos, se opta por un criterio de parada para residuos de orden de magnitud de  $10^{-5}$ . La distribución de valores de  $y^+$  sobre el perfil se observa en la siguiente figura.



Efectivamente se observa que los valores de  $y^+$  se encuentra dentro del rango admisible para funciones de pared estándar.

Finalmente, tras la resolución de cada caso, los valores de sustentación y resistencia (lift y drag), se configuran como parámetros de salida del problema y son guardados en un archivo de texto, mediante otro script en Python.

## EL PAPEL DE MANDO DE *MATLAB* Y LA TAREA DE OPTIMIZACIÓN

Las tareas descritas anteriormente deben seguir un determinado orden de ejecución. Para ello, se crea en *MATLAB* un script que se encarga de ir ejecutando cada una de dichas tareas en el orden necesario. En primer lugar, la geometría creada con el código C++ se importa en *MATLAB* a través del archivo *.dat* generado, pudiendo con ello ser representado en 2-D. La siguiente tarea es iniciar el ANSYS Workbench mediante el comando DOS de *MATLAB*, en modo *batch*, que a su vez es controlado por el script en Python ya comentado.

En cuanto a la optimización, se utiliza la herramienta *Optimization Toolbox* que contiene *MATLAB*. Dicha herramienta consta de diversos algoritmos que permiten minimizar/maximizar una determinada función objetivo. Presenta una interfaz gráfica que permite elegir el tipo de algoritmo a emplear, el punto inicial, las restricciones, condiciones de contorno, criterios de parada, etc.

El algoritmo elegido es el llamado *Pattern Search*, y debe escogerse un punto inicial, y como restricción, que la coordenada x del segundo punto debe ser siempre mayor que la del primero. Como límites, se toma:

$$0 \leq x_1 \leq 0.3; 0 \leq y_1 \leq 0.2; 0 \leq x_2 \leq 0.6; 0 \leq y_2 \leq 0.2; 1.9 \leq L \leq 2$$

La misión de dichos límites es evitar que la curvatura del perfil se vuelva demasiado acentuada, resultando en un pobre rendimiento de dicho perfil.

El algoritmo *Pattern Search* no requiere ninguna información sobre el gradiente de la función objetivo; simplemente compara valores discretos de dicha función. En este caso es la única opción puesto que no se tiene expresión matemática que defina la función objetivo, si no que la función objetivo es simplemente la función de MATLAB que controla todo el proceso y que en última instancia, imprime el valor de la fuerza de sustentación por pantalla.

Este algoritmo, a partir de un punto inicial, crea un patrón de puntos alrededor del mismo, y en dichos puntos evalúa la función objetivo. Si alguno de los puntos consigue una mejora (es decir, un mayor valor de *lift*), el punto que mayor mejora suponga, se convertirá en el punto a partir del cual se construirá el patrón de puntos en la siguiente iteración. Si no se consigue mejorar la función, simplemente se crea otro patrón de puntos más cercanos al punto central, y se vuelve a evaluar, y así sucesivamente.

Dependiendo del método de elección del patrón elegido, dicho patrón será uno u otro. En este caso se opta por *GSS Positive Basis 2N*, que crea un patrón formado por un número de vectores igual al doble del número de variables independientes del problema. Las variables son, como ya se ha comentado, las coordenadas x e y de los dos puntos que definen la curvatura del *spline*, más el valor de la longitud del perfil (puesto que se parte de un problema adimensional); es decir, un total de 5 variables independientes. Por lo tanto, se tiene un patrón de 10 vectores, a saber:

[1 0 0 0 0]; [0 1 0 0 0]; [0 0 1 0 0]; [0 0 0 1 0]; [0 0 0 0 1];

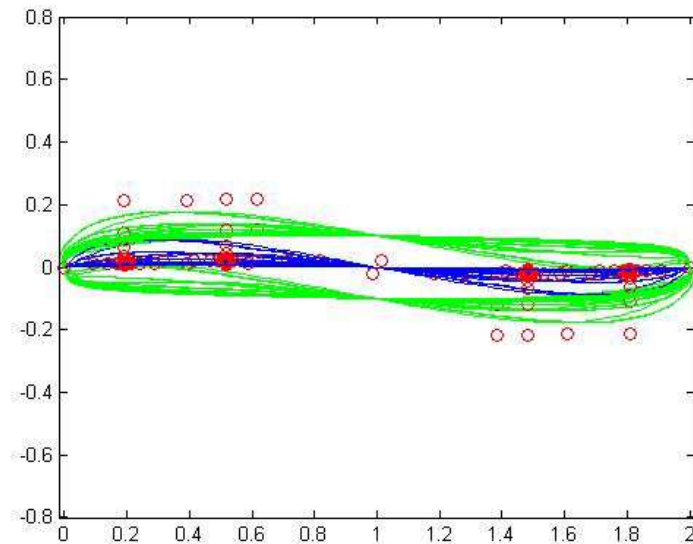
[-1 0 0 0 0]; [0 -1 0 0 0]; [0 0 -1 0 0]; [0 0 0 -1 0]; [0 0 0 0 -1]

Otro parámetro a tener en cuenta en el tamaño de la malla, que representa la longitud de los vectores, es decir, la distancia de los puntos del patrón respecto al punto central. La malla se expande (es decir, el tamaño de malla aumenta), si la búsqueda de un punto del patrón que mejore la función objetivo es satisfactoria. Por el contrario, si el patrón actual no es capaz de encontrar un punto donde la función objetivo sea mayor (si se busca maximizar), la malla se contrae.

Según el tamaño de malla se vaya haciendo más y más pequeño, cuando la búsqueda se encuentre cercana al punto óptimo deseado y el algoritmo no sea capaz de encontrar un mejor valor de la función, llegado a un determinado criterio de parada (en este caso, tamaño de malla más pequeño de orden  $10^{-3}$ ) el proceso se detendrá, y el valor de función obtenido será el máximo, y el punto será aquel que maximiza dicho valor.

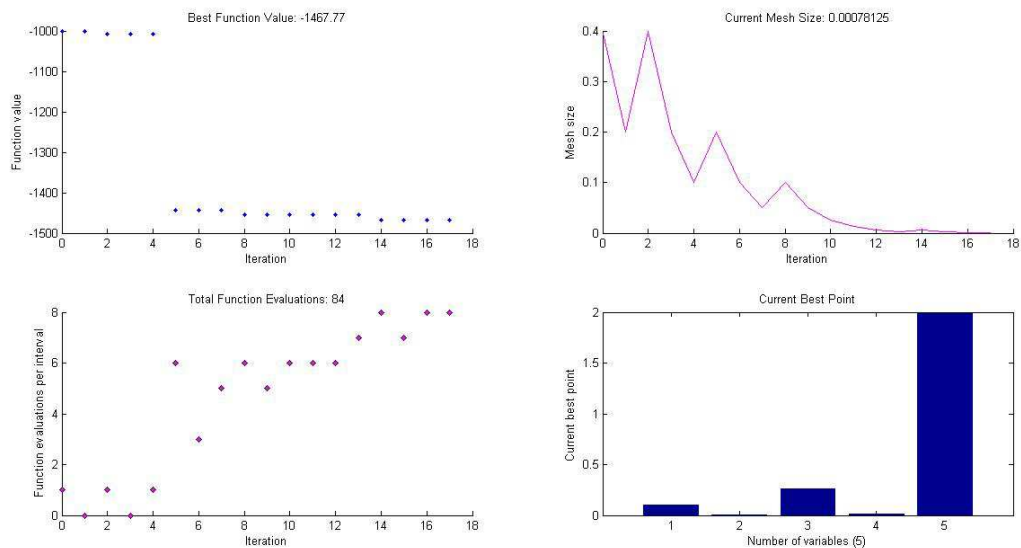
## RESULTADOS Y DISCUSIÓN

Partiendo de un determinado punto inicial (aquel que define las coordenadas de los puntos en el perfil en forma de S descrito al principio del documento), la evolución de la geometría de los diferentes perfiles a lo largo de la optimización puede observarse en la figura que aparece a continuación.



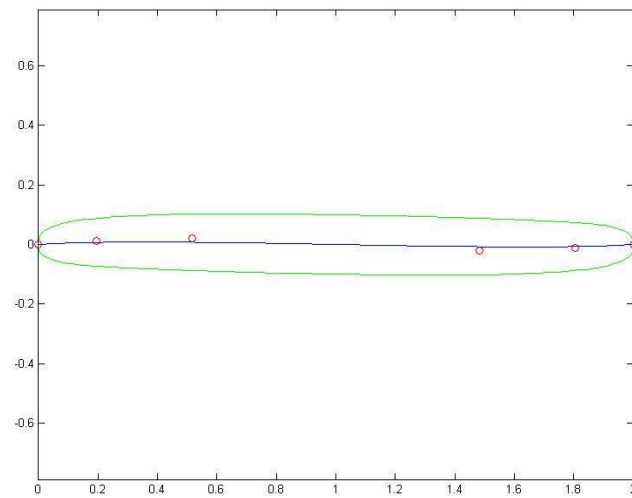
Puede observarse como, a pesar de partir de un perfil en forma de S, la optimización tiende poco a poco hacia perfiles de geometría más aplanada, que ofrecen un mejor rendimiento.

En la siguiente figura se muestra de manera conjunta la evolución de cada uno de los diferentes parámetros de la optimización a lo largo de la misma.



De izquierda a derecha y de arriba abajo: primero se muestra la evolución del valor de la función objetivo (fuerza sustentadora), a continuación la variación del tamaño de la malla, en tercer lugar el número de veces que ha sido evaluada la función en puntos del patrón en cada iteración, y finalmente un diagrama de barras que muestra el valor de las variables independientes del problema (las cinco ya descritas) a la finalización del proceso de optimización.

La siguiente figura muestra el perfil obtenido en *MATLAB*, al que corresponde el máximo valor de *lift* una vez finalizada la optimización.



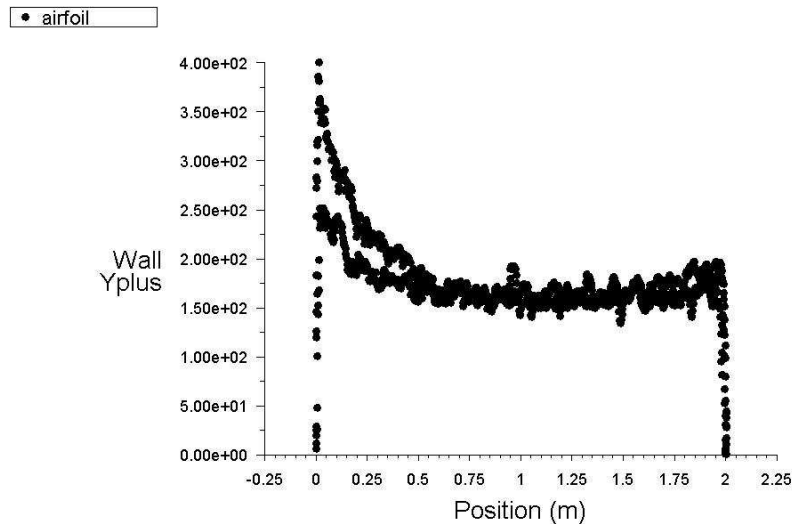
Puede apreciarse como, efectivamente, y tal y como se ha comentado previamente, el perfil óptimo presenta una geometría más bien plana, en lugar de la forma en S inicial. Sin embargo, para el cometido principal, que es operar corrientes oscilantes derivadas de las mareas, puesto que sigue presentando la condición de simetría, dicho perfil plano es perfectamente válido.

Los valores tanto de las variables independientes como de lift y drag para el perfil óptimo se resumen en la siguiente tabla.

$x_1$	$y_1$	$x_2$	$y_2$	Longitud (m)	Lift (N)	Drag (N)	lift/drag
0,098	0,006	0,258	0,0100	2	1467,77	320,88	4,574

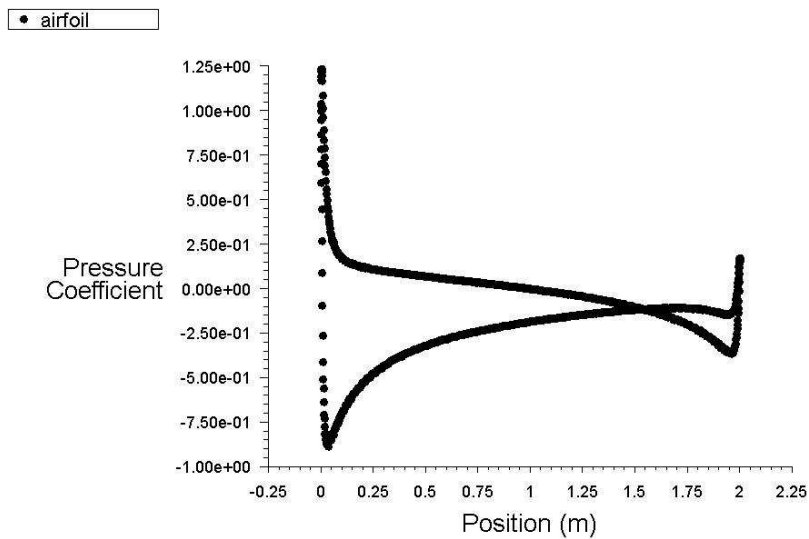
Una vez más, comprobando la fiabilidad de la malla a la hora de resolver el campo fluido alrededor de la pared del perfil, se muestra el  $y^+$  a lo largo de la superficie del álabe en la siguiente figura.





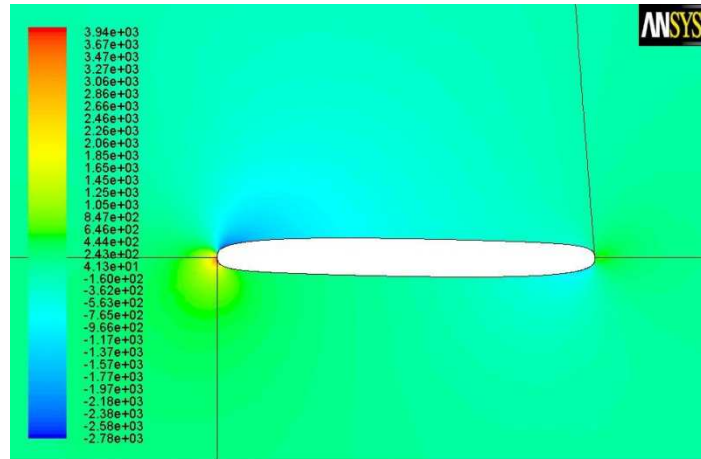
Una vez más, los valores se encuentran comprendidos en el intervalo aceptable para la resolución de la función de pared.

Por otro lado, la distribución del coeficiente de presión,  $C_p$ , se muestra en la siguiente imagen.

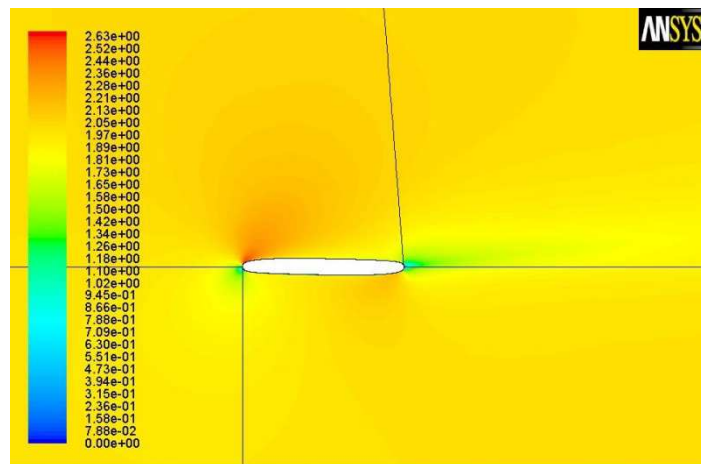


La curva superior corresponde al intradós, mientras que la inferior, al extradós. Extrayendo el valor medio de la integral de superficie de  $C_p$ , este resulta en -0,122, lo que indica que, a pesar de la alta sobrepresión en el borde de ataque, la contribución total del efecto de succión es más importante de el de dicha sobrepresión.

A continuación, se muestran varias figuras que representan el campo de presiones y de velocidades alrededor del perfil

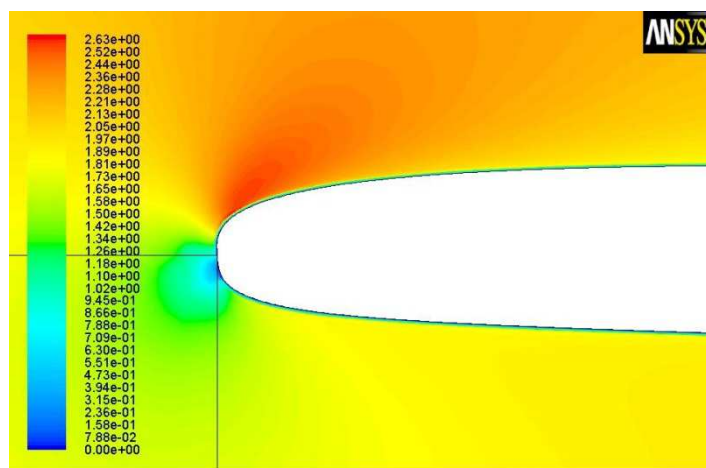


Contours of Static Pressure (pascal)



Contours of Velocity Magnitude (m/s)

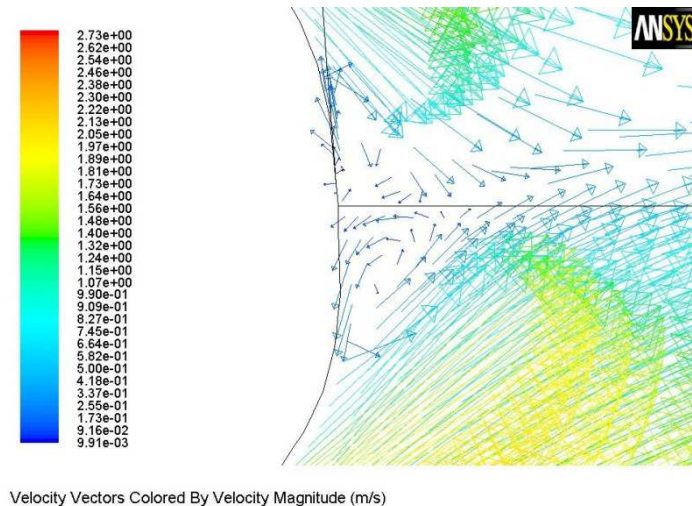
Haciendo zoom cerca del borde de ataque para mayor nivel de detalle:



Contours of Velocity Magnitude (m/s)

En lo que respecta al campo de velocidades, se observa como, debido al pequeño ángulo de ataque y a la geometría plana, la estela tras el perfil es estrecha y de corta longitud, perdiéndose la mayor parte de su efecto a poca distancia aguas abajo.

A continuación aparece una figura con el campo de vectores velocidad en el campo fluido cercano al borde de salida del perfil.



Se observa como, a pesar del pequeño ángulo de ataque ( $5^\circ$ ), y debido al perfil redondeado en el borde de salida, aparece una muy pequeña zona de recirculación.

## CONCLUSIÓN Y TRABAJOS FUTUROS

La principal conclusión extraíble de este trabajo es que, a pesar de partir de un modelo con geometría en forma de S, con el transcurso de la optimización, se tiende poco a poco hacia geometrías más suavizadas, aplanadas. Cuanto más acentuada es la curvatura de la línea media, y por lo tanto, del perfil, las prestaciones del álabe caen de manera considerable.

Como posibles trabajos futuros, se enumeran una serie de ellos a continuación:

- Utilización de un algoritmo multipropósito, con la finalidad de adicionalmente tratar de minimizar el *drag* o maximizar el cociente *lift/drag*.
- Comparación con otros modelos de turbulencia, y validación en túnel de viento que permita la elección del modelo más fiable y próximo a los datos experimentales.
- Optimización de varias secciones 2D del álabe, consistentes en perfiles simétricos, y con ello poder estudiar el problema en 3D. Utilizar para ello, por ejemplo, CFX.
- Estudio de la influencia de otros parámetros (velocidad de entrada, propiedades del fluido como temperatura, densidad, viscosidad, etc).